COMPUTER NETWORKS

Assignment 3 Report

22CS10025

G SAI SHASANK

Note:

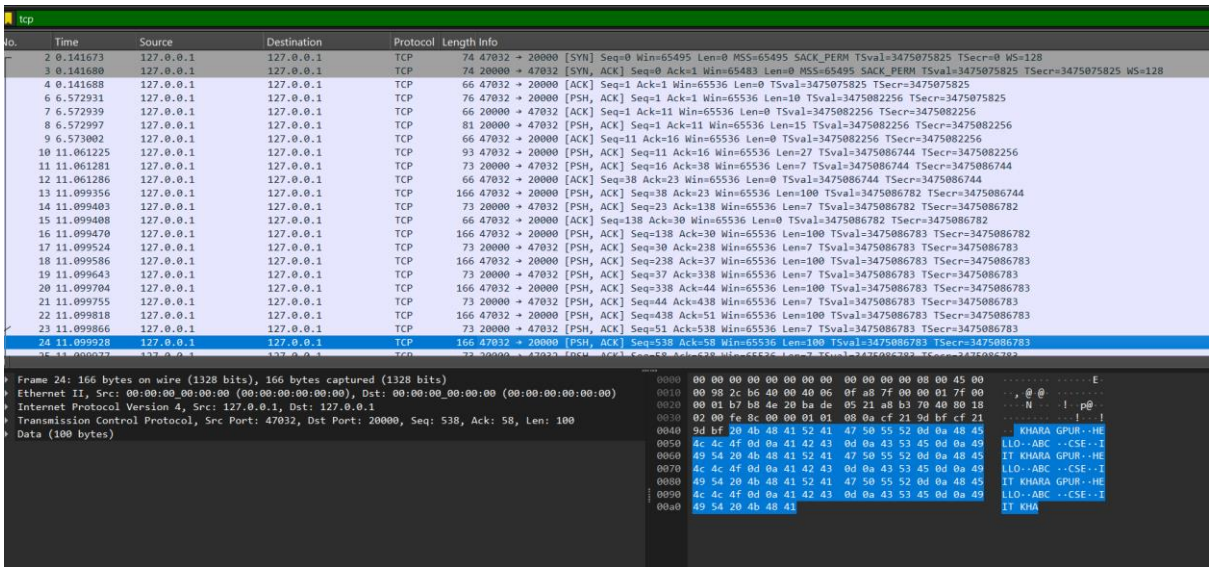
- This analysis is done with the input file as file3.txt.
- At the time of analysis, the buffer size at the client side was 100 bytes and at the server side was 100 bytes.
- Except for the packets of the file data transfer and three-way handshake, you can see some more packets in the pcap files. These packets are for differentiating between process/acknowledgment of a process completion/start.

Q1: What are the source and destination IP addresses and ports? Share the screenshots to justify your answer.

A1: (While the data transfer takes place from the client to server and vice versa for the case where the data transfer takes place from the server to the client)

- Source IP address: 127.0.0.1
- Destination IP address: 127.0.0.1
- Source Port No: 47032
- Destination Port No: 20000

Associated Screanshot:



Note: No packet will get the size more than 166(100 for data and 66 for other information of packet)

Q2: Inspect the Three-way handshaking procedure and capture all packets exchanged for it. Attach the necessary screenshots to demonstrate it.

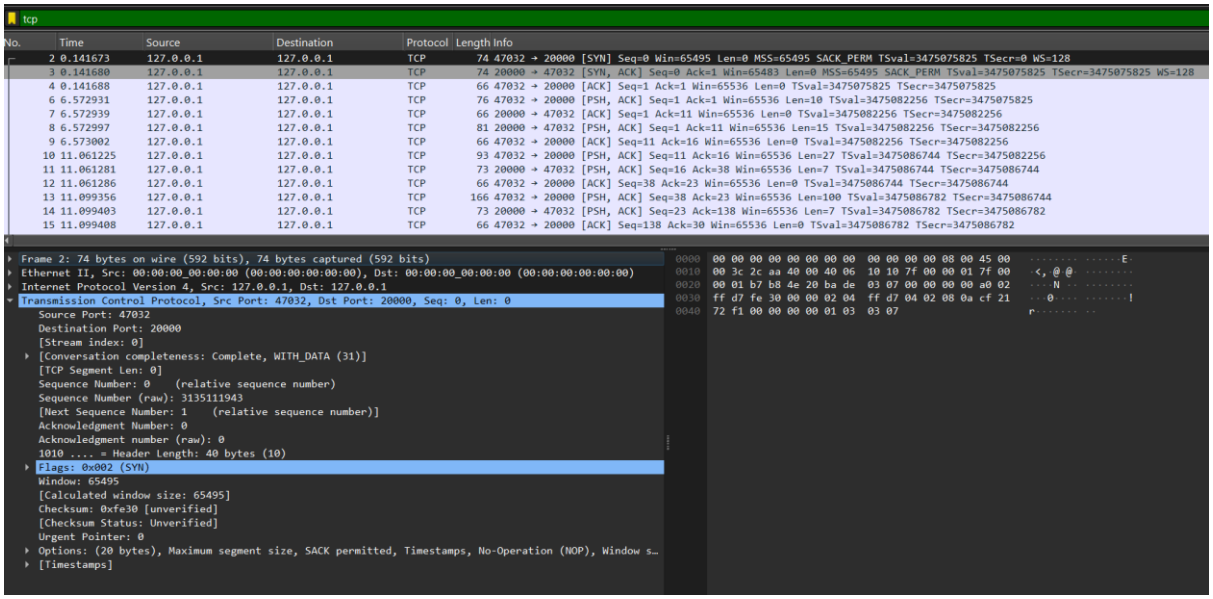
A2: In the above question's screenshot, the first 3 packets belong to the three-way handshake process.

In a Three-way handshake process:

Step 1: SYN (Synchronize)

The client initiates the connection by sending a TCP packet to the server with the SYN (Synchronize) flag set. This packet contains an initial sequence number (ISN) chosen by the client.

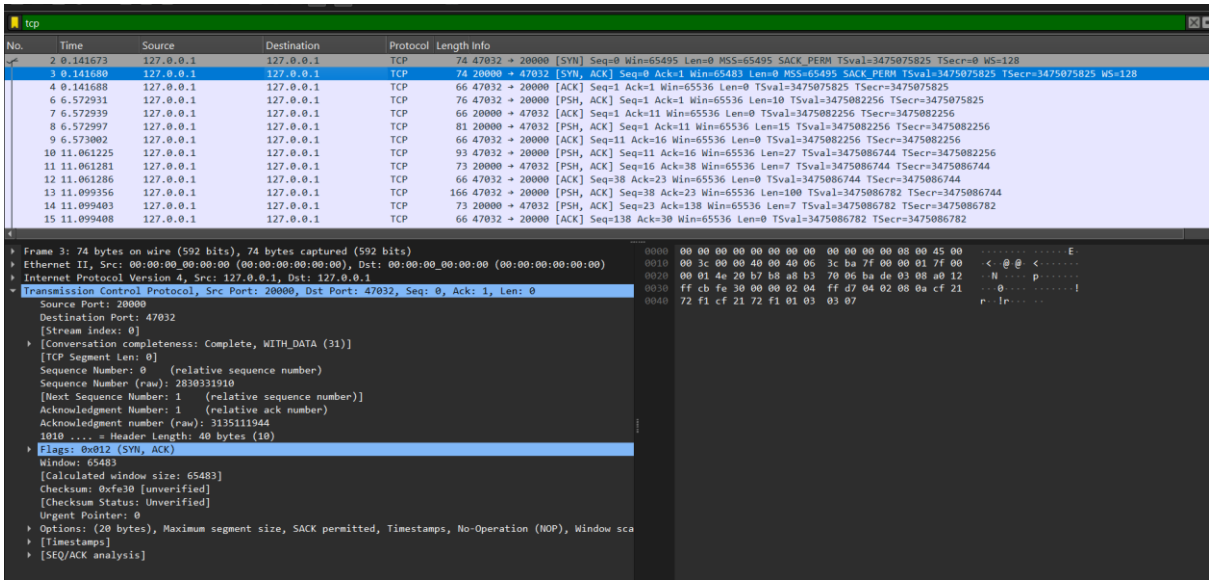
- Here the ISN from the client side is 2830332701.
- **Associated screenshot**:



Step 2: SYN-ACK (Synchronize-Acknowledge)

Upon receiving the SYN packet, the server responds with its own TCP packet. This packet has both the SYN and ACK flags set, acknowledging the client's SYN and proposing its own initial sequence number (ISN).

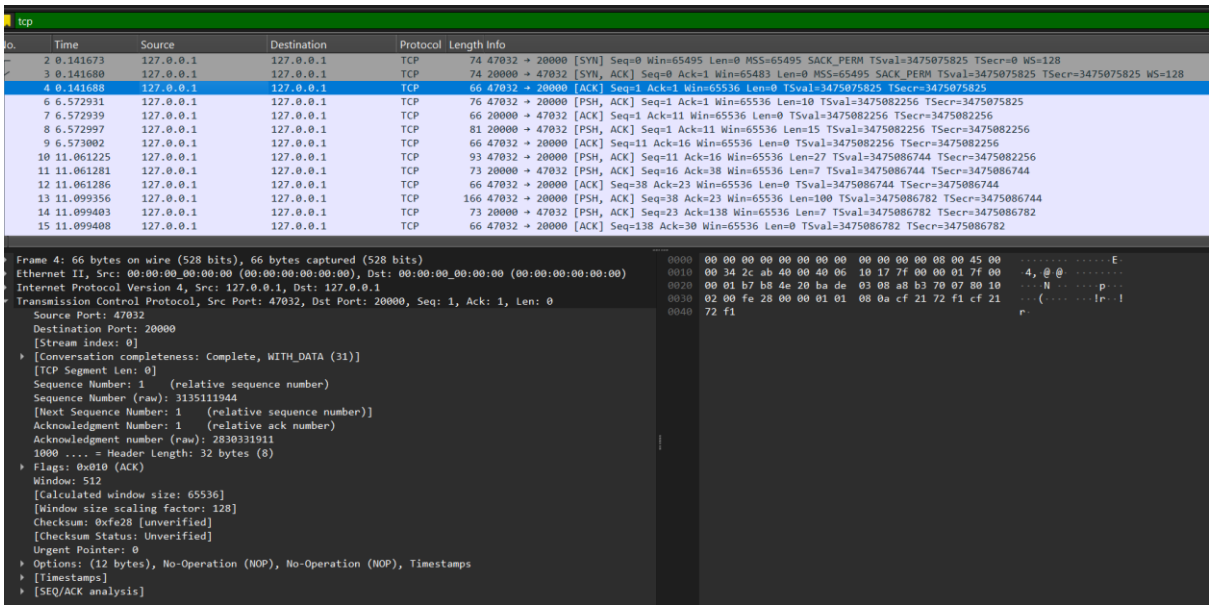
- Here the ISN from the server side is 3135112750.
- **Associated screenshot**:



Step 3: ACK (Acknowledge)

The client then sends a final TCP packet to the server with the ACK flag set, acknowledging the server's SYN. At this point, both the client and server have established a reliable, full-duplex communication channel.

- You can see in the screenshot that the ACK flag is set.
- **Associated screenshot**:



Q3: Inspect the connection closure procedure and capture all packets exchanged for it. Attach the necessary screenshots to demonstrate it.

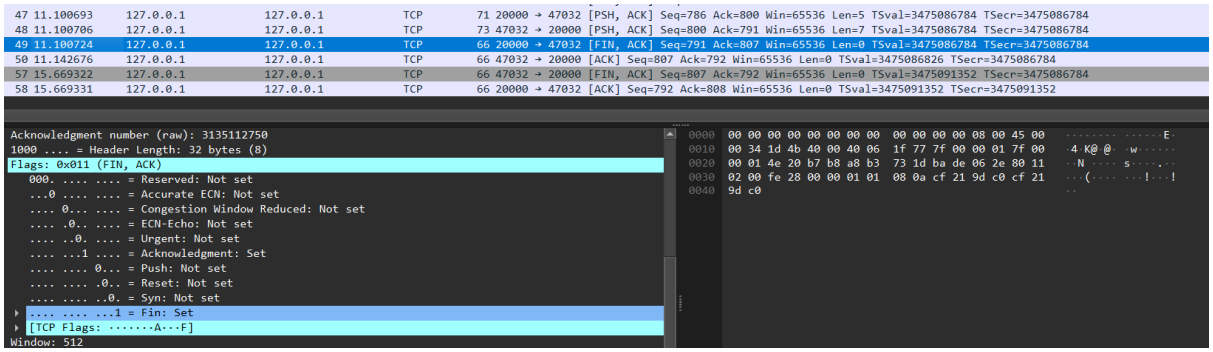
A3: The last 3 packets in the first question's screenshot are the packets responsible for the connection closure procedure.

The TCP connection closing procedure using 3 packets involves:

1. **FIN (Finish)**

   One party (either client or server) initiates the connection closure by sending a FIN packet. This packet has the FIN flag set to '1' and indicates that the sender has no more data to transmit.

   - In the screenshot below, we can see that the client starts the connection termination process.
   - **Associated screenshot:**



2. FIN-ACK

Upon receiving the FIN packet, the other party responds with a FIN-ACK packet. This packet acknowledges the receipt of the FIN and also indicates that this party is ready to close its side of the connection.

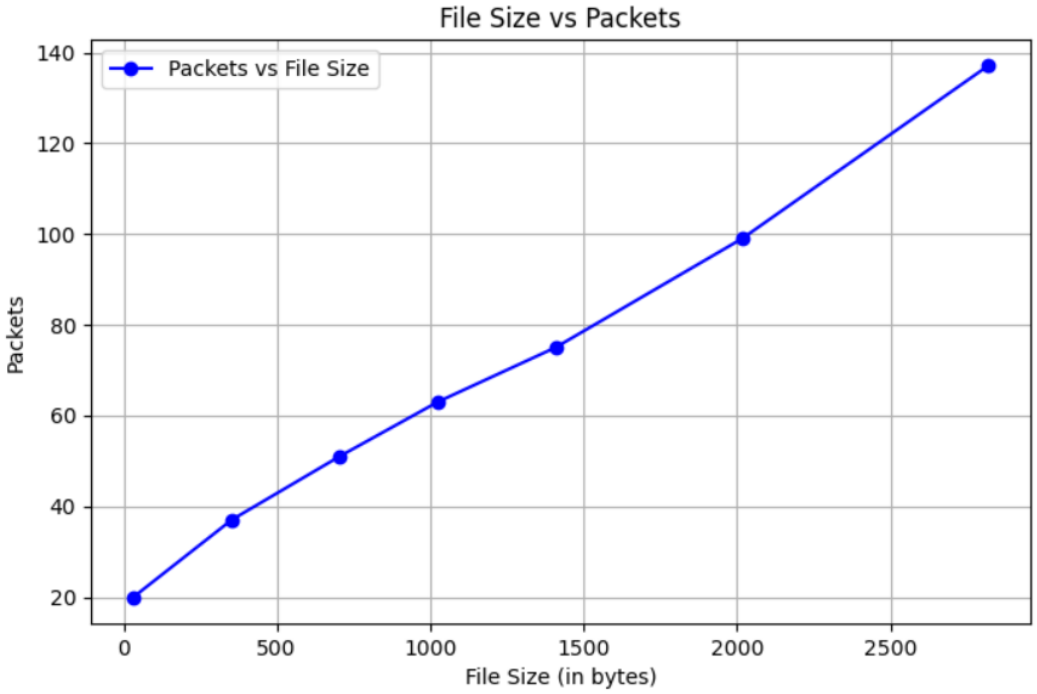
3. ACK (Acknowledge)

The party that initiated the closure sends a final ACK packet to acknowledge the receipt of the FIN-ACK. This completes the connection termination process.

Q4: Inspect the traffic and count the number of packets exchanged for the transfer of a file (related to data only) between client and server. Plot a graph 'file size vs the number of packets' based on your observation.

A4:

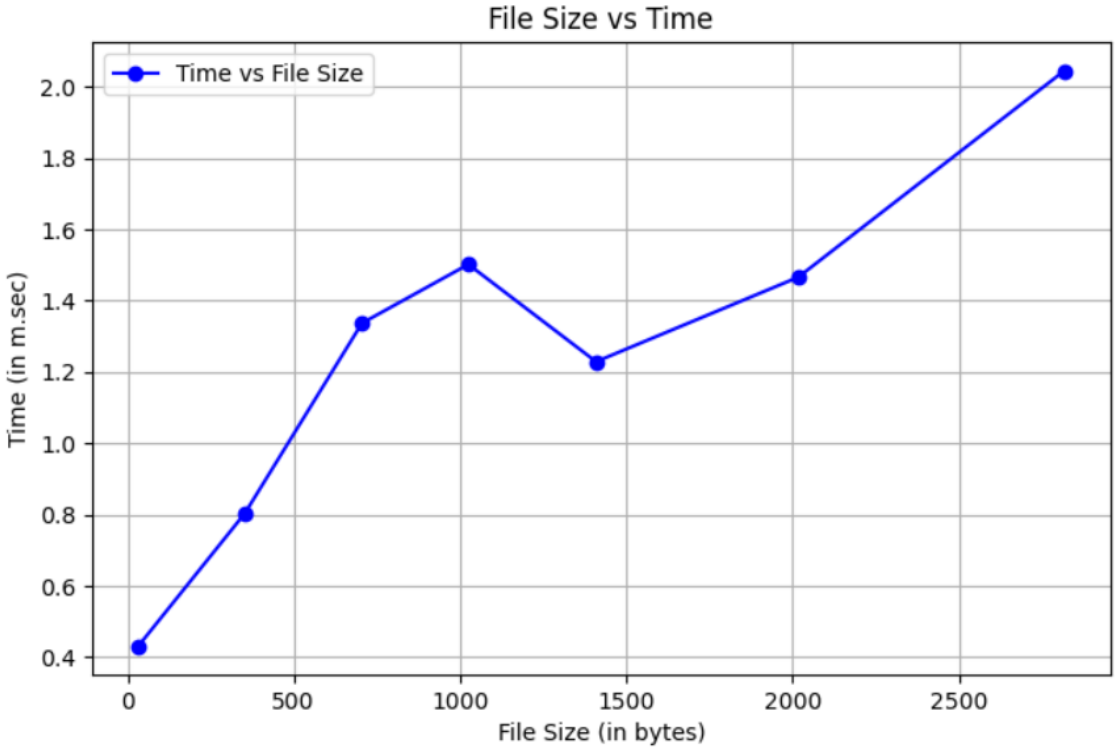
File size(in bytes)	Packets
33	20
353	37
705	51
1025	63
1409	75
2017	99
2817	137



Q5: Measure the total time taken for the file transfer, its encryption, and sending it back from the server to the client. Plot a graph 'file size vs time' based on your observation and attach the necessary screenshots.

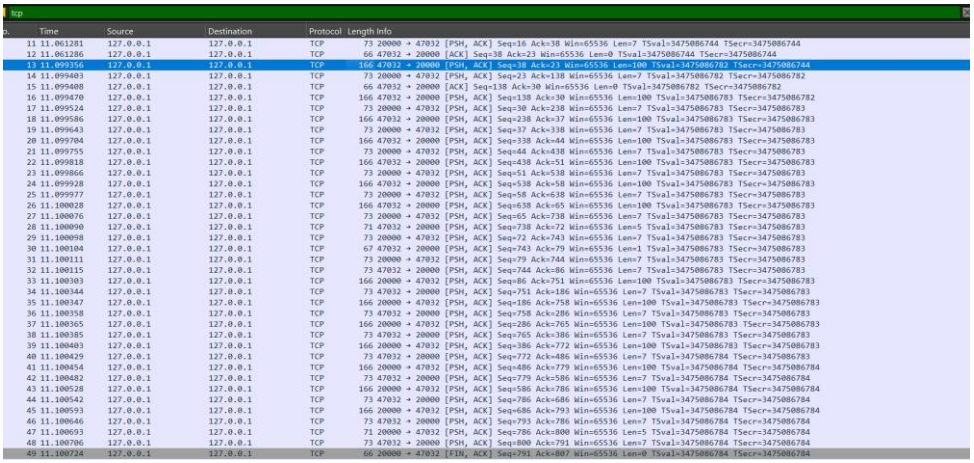
A5:

File size(in bytes)	Time (in m.sec)
33	0.432
353	0.804
705	1.337
1025	1.502
1409	1.229
2017	1.466
2817	2.045



● Total file transfer time from client to server: **1.337ms** (file size: 705 bytes).

● **Associated screenshot:**



No.	Time	Source	Destination	Protocol	Length	Info
11	11.061281	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=16 Ack=38 Win=65536 Len=7 TSval=3475086744 TSecr=3475086744
12	11.061286	127.0.0.1	127.0.0.1	TCP	66	47032 → 20000 [ACK] Seq=38 Ack=23 Win=65536 Len=0 TSval=3475086744 TSecr=3475086744
13	11.099356	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=38 Ack=23 Win=65536 Len=100 TSval=3475086782 TSecr=3475086744
14	11.099403	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=23 Ack=138 Win=65536 Len=7 TSval=3475086782 TSecr=3475086782
15	11.099408	127.0.0.1	127.0.0.1	TCP	66	47032 → 20000 [ACK] Seq=138 Ack=30 Win=65536 Len=0 TSval=3475086782 TSecr=3475086782
16	11.099470	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=138 Ack=30 Win=65536 Len=100 TSval=3475086783 TSecr=3475086782
17	11.099524	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=30 Ack=230 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
18	11.099586	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=230 Ack=37 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
19	11.099643	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=37 Ack=330 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
20	11.099704	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=330 Ack=44 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
21	11.099755	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=44 Ack=430 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
22	11.099818	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=430 Ack=51 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
23	11.099866	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=51 Ack=530 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
24	11.099928	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=530 Ack=58 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
25	11.099977	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=58 Ack=630 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
26	11.100028	127.0.0.1	127.0.0.1	TCP	166	47032 → 20000 [PSH, ACK] Seq=630 Ack=65 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
27	11.100076	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=65 Ack=730 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
28	11.100090	127.0.0.1	127.0.0.1	TCP	71	47032 → 20000 [PSH, ACK] Seq=730 Ack=72 Win=65536 Len=5 TSval=3475086783 TSecr=3475086783
29	11.100098	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=72 Ack=743 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
30	11.100104	127.0.0.1	127.0.0.1	TCP	67	47032 → 20000 [PSH, ACK] Seq=743 Ack=79 Win=65536 Len=1 TSval=3475086783 TSecr=3475086783
31	11.100111	127.0.0.1	127.0.0.1	TCP	73	20000 → 47032 [PSH, ACK] Seq=79 Ack=744 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
32	11.100115	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=744 Ack=86 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
33	11.100303	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=86 Ack=751 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
34	11.100344	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=751 Ack=186 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
35	11.100347	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=186 Ack=758 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
36	11.100358	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=758 Ack=286 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
37	11.100365	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=286 Ack=765 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
38	11.100385	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=765 Ack=386 Win=65536 Len=7 TSval=3475086783 TSecr=3475086783
39	11.100403	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=386 Ack=772 Win=65536 Len=100 TSval=3475086783 TSecr=3475086783
40	11.100429	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=772 Ack=486 Win=65536 Len=7 TSval=3475086784 TSecr=3475086783
41	11.100454	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=486 Ack=779 Win=65536 Len=100 TSval=3475086784 TSecr=3475086784
42	11.100482	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=779 Ack=586 Win=65536 Len=7 TSval=3475086784 TSecr=3475086784
43	11.100528	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=586 Ack=786 Win=65536 Len=100 TSval=3475086784 TSecr=3475086784
44	11.100542	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=786 Ack=686 Win=65536 Len=7 TSval=3475086784 TSecr=3475086784
45	11.100593	127.0.0.1	127.0.0.1	TCP	166	20000 → 47032 [PSH, ACK] Seq=686 Ack=793 Win=65536 Len=100 TSval=3475086784 TSecr=3475086784
46	11.100646	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=793 Ack=786 Win=65536 Len=7 TSval=3475086784 TSecr=3475086784
47	11.100693	127.0.0.1	127.0.0.1	TCP	71	20000 → 47032 [PSH, ACK] Seq=786 Ack=800 Win=65536 Len=5 TSval=3475086784 TSecr=3475086784
48	11.100706	127.0.0.1	127.0.0.1	TCP	73	47032 → 20000 [PSH, ACK] Seq=800 Ack=791 Win=65536 Len=7 TSval=3475086784 TSecr=3475086784
49	11.100724	127.0.0.1	127.0.0.1	TCP	66	20000 → 47032 [FIN, ACK] Seq=791 Ack=807 Win=65536 Len=0 TSval=3475086784 TSecr=3475086784

Q6: Calculate the average packet size exchanged during data communication? Take reference from the plotted graph in the previous question.

A6:

Average packet size (approximately): (calculated based on file1 and file 7)

= (y2-y1)/(x2-x1)

= (2817-33)/(137-21)

= 24 bytes of data per packet